

Parametrized empirical interpolation for discrete implicit evolution operators

- ▶ Extending reduced basis methods
- ▶ for **parametrized** evolution problems

$$\partial_t u(x, t; \mu) - \mathcal{L}(\mu)[u(x, t; \mu)] = 0$$

- ▶ discretized by **first order** numerical schemes.
- ▶ **Focus:** **non-linear implicit** discretizations with Newton schemes.
- ▶ **Main ingredient:** **empirical interpolation** of discrete operators and its Fréchetderivatives.



Outline

- ▶ Reduced basis method in pictures
- ▶ Empirical interpolation of discrete operators
- ▶ Reduced basis scheme with non-linear implicit operators
- ▶ A posteriori error estimation
- ▶ Numerical results and validation
- ▶ Software concept (RBmatlab/dune-rb interface)

Outline

- ▶ Reduced basis method in pictures
- ▶ **Empirical interpolation of discrete operators**
- ▶ Reduced basis scheme with non-linear implicit operators
- ▶ A posteriori error estimation
- ▶ Numerical results and validation
- ▶ Software concept (RBmatlab/dune-rb interface)



Setting

- ▶ parametrized PDEs with parameters for geometry, material, control,...
- ▶ applications relying on **rapid** or many **repeated** computations for different parameters.



Setting

- ▶ parametrized PDEs with parameters for geometry, material, control,...
- ▶ applications relying on **rapid** or many **repeated** computations for different parameters.

Ideas

- ▶ Generation of reduced basis space
- ▶ Offline/Online decomposition
- ▶ A posteriori error estimates

Parametrised evolution equation

Analytical formulation

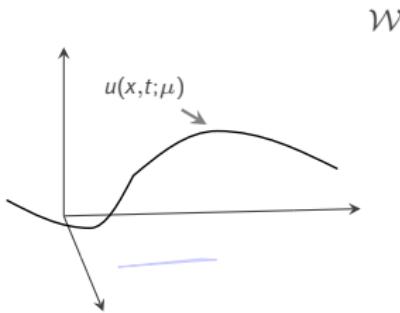
Solve

$$\begin{aligned} u(x, o; \mu) &:= u_0(x; \mu) \\ \partial_t u(x, t; \mu) - \mathcal{L}[u(x, t; \mu)] &= o \end{aligned}$$

plus boundary conditions. Solutions $u(\cdot, t; \mu)$ live in a (Sobolev) space \mathcal{W} for each $t \in [o, T_{\max}]$, $\mu \in \mathcal{P}$.

Notations

$$\begin{array}{l} \mathcal{L} \\ \mathcal{W} \\ \hline u(x, t; \mu) \end{array}$$



Discrete simulations (FV, FE, DG, ...)

Discretization

Solve

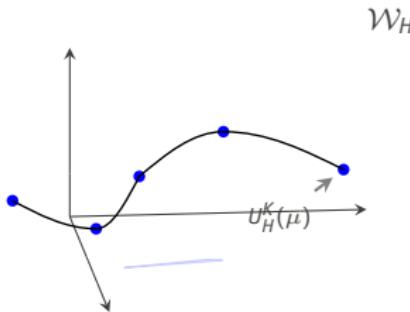
$$u_h^0(\mu) := P[u_0(x; \mu)]$$

$$L_I(\mu, \Delta t^k)[u_h^{k+1}] - L_E(\mu, \Delta t^k)[u_h^k] = 0,$$

with $\Delta t = \frac{T_{\max}}{K}$. Solutions build trajectories in **high** dimensional **discrete** function space \mathcal{W}_H .

Notations

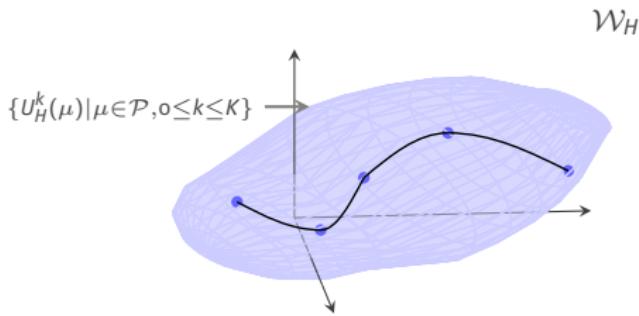
\mathcal{L}
\mathcal{W}
$u(x, t; \mu)$
L_I, L_E, P
\mathcal{W}_H
$\{u_h^k(\mu)\}_{k=1}^K$



Discrete simulations (FV, FE, DG, ...)

Discretization

Because of the parametrized form of the problem, we get a manifold of solutions $\{u_h^k | \mu \in \mathcal{P}, 0 \leq k \leq K\} \subset \mathcal{W}_H$. This can be approximated by a linear subspace $\mathcal{W}_N := \text{span}\{\varphi_i\}_{i=1}^N$.



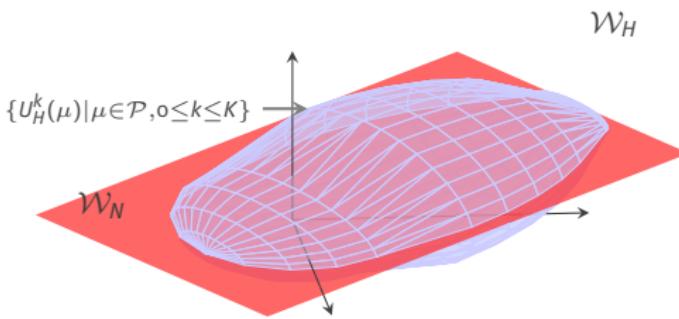
Notations

\mathcal{L}
\mathcal{W}
$u(x, t; \mu)$
L_I, L_E, P
\mathcal{W}_H
$\{u_h^k(\mu)\}_{k=1}^K$

Discrete simulations (FV, FE, DG, ...)

Discretization

Because of the parametrized form of the problem, we get a manifold of solutions $\{u_h^k | \mu \in \mathcal{P}, 0 \leq k \leq K\} \subset \mathcal{W}_H$. This can be approximated by a linear subspace $\mathcal{W}_N := \text{span}\{\varphi_i\}_{i=1}^N$.



Notations

\mathcal{L}
\mathcal{W}
$u(x, t; \mu)$
L_I, L_E, P
\mathcal{W}_H
$\{u_h^k(\mu)\}_{k=1}^K$

Reduced simulations (Projection on reduced basis space)

Reduced formulation

Solve

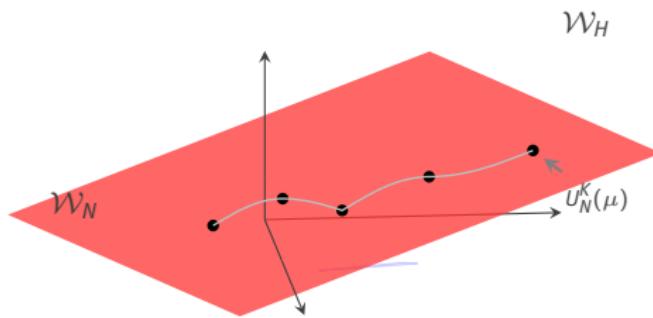
$$\mathbf{a}^0(\mu) := \mathbf{P}[u_0(\mu)]$$

$$\mathbf{L}_I(\mu, \Delta t^k)[\mathbf{a}^{k+1}] - \mathbf{L}_E(\mu, \Delta t^k)[\mathbf{a}^k] = \mathbf{o}.$$

Solutions $\mathbf{a}^k(\mu) := (a_n^k(\mu))_{n=1}^N$ are vectors in **low** dimensional space \mathbb{R}^N .

Notations

\mathcal{L}
\mathcal{W}
$u(x, t; \mu)$
L_I, L_E, P
\mathcal{W}_H
$\{u_h^k(\mu)\}_{k=1}^K$
$\mathbf{L}_I, \mathbf{L}_E, \mathbf{P}$
\mathcal{W}_N
$\{\mathbf{a}^k(\mu)\}_{k=1}^K$



Reduced simulations (Projection on reduced basis space)

Reconstruction

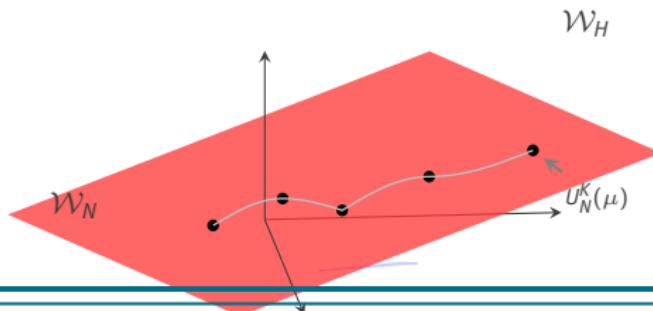
Reconstruct the solution

$$u_N^k(\mu) := \sum_{n=1}^N a_n^k(\mu) \varphi_n \in \mathcal{W}_N$$

in reduced basis space \mathcal{W}_N or preferably: Evaluate an output functional $s(\mu)$.

Notations

\mathcal{L}
\mathcal{W}
$u(x,t;\mu)$
L_I, L_E, P
\mathcal{W}_H
$\{u_h^k(\mu)\}_{k=1}^K$
L_I, L_E, P
\mathcal{W}_N
$\{a^k(\mu)\}_{k=1}^K$
$\mathcal{W}_H, \mathcal{W}_O$
$u_N^k(\mu), s(\mu)$





Ingredients during offline phase

1. Generation of reduced basis space \mathcal{W}_N .
2. Projection of operators L_I, L_E, P onto RB space resulting in reduced matrices $\mathbf{L}_I, \mathbf{L}_E, \mathbf{P}$.

Ingredients during offline phase

1. Generation of reduced basis space \mathcal{W}_N .
2. Projection of operators L_I, L_E, P onto RB space resulting in reduced matrices $\mathbf{L}_I, \mathbf{L}_E, \mathbf{P}$.

Ingredients during online phase

1. Efficient computation of reduced scheme independently of high dimensional ingredients.
2. Assuring and measuring reliability of reduced output. (a posteriori error estimates)

Reduced basis generation

POD-greedy algorithm

- ▶ **Initialization:** Define an initial reduced basis \mathcal{W}_{N_0} and a finite training parameter set $M_{\text{train}} \subset \mathcal{P}$.

Reduced basis generation

POD-greedy algorithm

- ▶ **Initialization:** Define an initial reduced basis \mathcal{W}_{N_0} and a finite training parameter set $M_{\text{train}} \subset \mathcal{P}$.
- ▶ **Iterative extension of basis:**
 - ▶ Find $\mu_{\max} := \arg \max_{\mu \in M_{\text{train}}} \max_{k=0}^K \|u_N^k(\mu) - u_h^k(\mu)\|$ (with efficient a posteriori error estimates).
 - ▶ Add principal components of projection error of trajectory trajectory $\left\{ u_N^k - u_h^k \right\}_{k=1}^K$ to reduced basis.
- ▶ **Stop:** if error is small enough.

Affinely decomposed operators

Definition

An operator is called **affinely decomposed** if it can be written as a sum of products of

- ▶ **efficiently** computable parameter **dependent** and
- ▶ parameter **independent** parts with affine dependence on the space variable.

Affinely decomposed operators

Definition

An operator is called **affinely decomposed** if it can be written as a sum of products of

- ▶ efficiently computable parameter **dependent** and
- ▶ parameter **independent** parts with affine dependence on the space variable.

Example

An operator L_E has the form

$$L_E = \sum_{q=1}^{Q_E} \sigma^q(\mu) L^q$$

with L^q linear. Then \mathbf{L}_E can be efficiently assembled as

$$\mathbf{L}_E = \sum_{q=1}^{Q_E} \sigma^q(\mu) \mathbf{L}^q \quad \text{with} \quad (\mathbf{L}^q)_{nm} = \int_{\mathbb{R}^d} L^q[\varphi_m] \varphi_n \quad \text{for } n, m = 1, \dots, N.$$

Classification of evolution problems

Operator constraints

	L_I	L_E
1	linear in space, affinely μ -dependent	
2	$= \text{Id}$	localized
3	coercive, non-coercive, linear, non-linear	$= \text{Id}$
4	linear in space	localized or: linear in space, affinely μ -dependent
5	localized, nonlinear	

Classification of evolution problems

Operator constraints

	L_I	L_E
1	linear in space, affinely μ -dependent	
2	$= \text{Id}$	localized
3	coercive, non-coercive, linear, non-linear	$= \text{Id}$
4	linear in space	localized or: linear in space, affinely μ -dependent
5	localized, nonlinear	

- ▶ Case 1 is discussed in [Haasdonk & Ohlberger, 2008].
- ▶ Case 2 is discussed in [Haasdonk et al., 2008] and [Drohmann et al., 2009]
- ▶ Case 3 is discussed in [Grepl, 2005], [Grepl et al., 2008] and [Knezevic et al., 2009]
- ▶ **Our focus:** nonlinear and non-affinely decomposed implicit operators.

Empirical interpolation: Idea

General operator approximation

Approximate operator evaluations

$$L(\mu)[U] \approx \mathcal{I}_M \left[L(\mu) \left[u_h^k(\mu) \right] \right] = \sum_{m=1}^M y_m(\mu; u_h(\mu)) \xi_m,$$

where $y_m(\mu) : \mathcal{W}_{\mathcal{H}} \rightarrow \mathbb{R}$ are efficiently computable functionals and ξ_m collateral reduced basis functions.

Empirical interpolation: Idea

General operator approximation

Approximate operator evaluations

$$L(\mu)[U] \approx \mathcal{I}_M \left[L(\mu) \left[u_h^k(\mu) \right] \right] = \sum_{m=1}^M y_m(\mu; u_h(\mu)) \xi_m.$$

Empirical interpolation [Barrault et al, 2004]

- ▶ collateral reduced basis $\mathcal{W}_M^{\text{CRB}}$ spanned by operator evaluations
 $q_m = L(\mu_m) \left[u_h^{k_m}(\mu_m) \right], m = 1, \dots, M$ for **suitably chosen** parameters $\mu_m \in \mathcal{P}$ and time steps k_m
- ▶ coefficient functionals $y_m(\mu; U) = L(\mu)[U](x_m)$ are exact operator evaluations at **interpolation points** $(x_m)_{m=1}^M$.

Empirical interpolation: Idea

General operator approximation

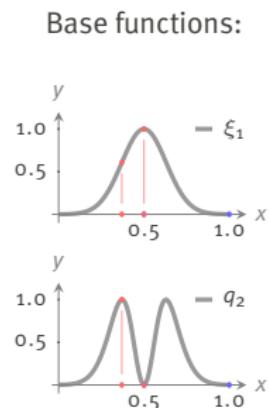
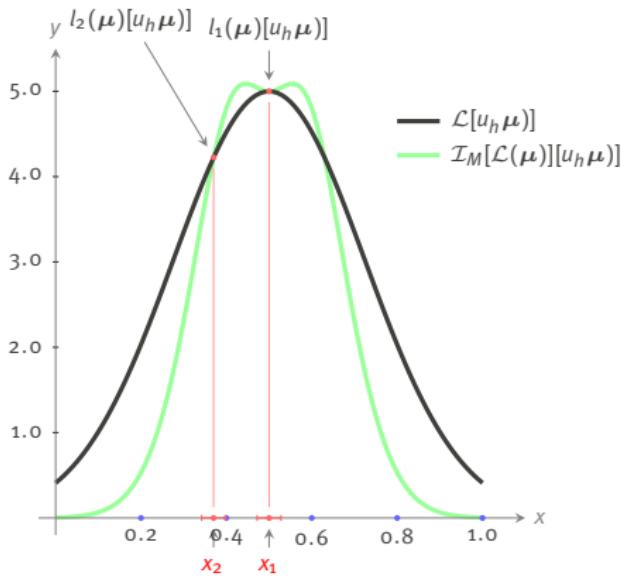
Approximate operator evaluations

$$L(\mu)[U] \approx \mathcal{I}_M \left[L(\mu) \left[u_h^k(\mu) \right] \right] = \sum_{m=1}^M y_m(\mu; u_h(\mu)) \xi_m.$$

Empirical interpolation [Barrault et al, 2004]

- ▶ collateral reduced basis $\mathcal{W}_M^{\text{CRB}}$ spanned by operator evaluations
 $q_m = L(\mu_m) \left[u_h^{k_m}(\mu_m) \right], m = 1, \dots, M$ for **suitably chosen** parameters $\mu_m \in \mathcal{P}$ and time steps k_m
- ▶ coefficient functionals $y_m(\mu; U) = L(\mu)[U](x_m)$ are exact operator evaluations at **interpolation points** $(x_m)_{m=1}^M$.
- ▶ CRB functions ξ_m are nodal in interpolation points, such that $\xi_m(x_n) = \delta_{nm}$.

Empirical interpolation: in pictures



Collateral reduced basis generation

CRB generation algorithm

- ▶ **Initialization:** Compute training set $L_{\text{train}} := \{L(\mu)[u_h^k(\mu)] | k = 0, \dots, K, \mu \in M_{\text{train}}\}$
- ▶ **Iterative extension of basis $m - 1 \rightarrow m$:**
 1. For all $v_h \in L_{\text{train}}$ find best approximation $v^* := \arg \min_{w \in \mathcal{W}_{m-1}^{\text{CRB}}} \|w - v_h\|.$
 2. Find worst approximated snapshot $v_m := \arg \max_{v_h \in L_{\text{train}}} \|v_h - v^*\|.$
 3. Compute residual of approximation r_m .
 4. Define new interpolation point and basis function: $x_m := \arg \sup_x |r_m(x)|, q_m := \frac{r_m}{r_m(x_m)}$
- ▶ **Stop:** if error is small enough.

Efficient online evaluations

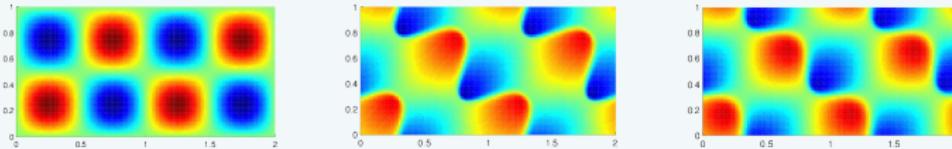
The coefficient functionals y_m can be computed efficiently during **online** phase, if

- ▶ operator has localized structure (**small stencil**) and
- ▶ local geometry information is precomputed in offline phase (local grid)

Example

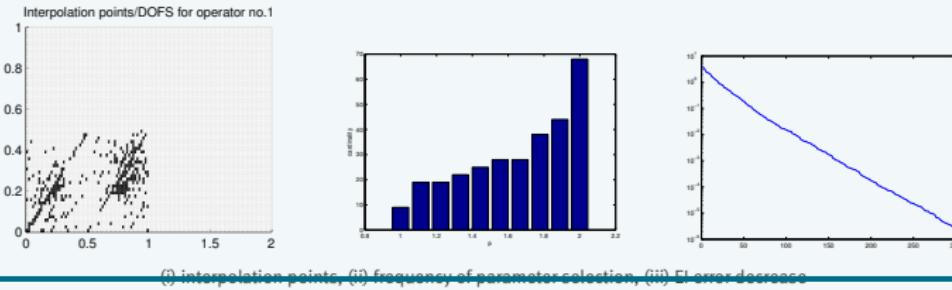
Burgers problem

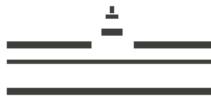
$\partial_t u - \nabla u^p = 0$ with periodic boundary conditions



(i) initial data snapshot ($T = 0$), (ii) snapshot for $p = 2$ at $T = 0.3$, (iii) snapshot for $p = 1.5$ at $T = 0.3$

Empirical interpolation of L_E :





Derivative of empirical interpolation

Observation

$$\mathcal{I}_M [\mathbf{D}L(u_h)] [v_h] = \sum_{m=1}^M \mathbf{D}y_m(u_h) [v_h] \xi_m$$

Derivative of empirical interpolation

Observation

$$\mathcal{I}_M [\mathbf{D}L(u_h)] [v_h] = \sum_{m=1}^M \mathbf{D}y_m(u_h) [v_h] \xi_m$$

Definition

- Dof mapping $\Phi : \mathbb{R}^H \rightarrow \mathcal{W}_H$, $\mathbf{U} := \{U_i\}_{i=1}^H \mapsto \sum_{i=1}^H U_i \psi_i$.
- Discrete operator $\bar{L} : \mathbb{R}^H \rightarrow \mathcal{W}_H$ given by $\bar{L}[\mathbf{U}] := L \circ \Phi[\mathbf{U}]$.

Derivative of empirical interpolation

Observation

$$\begin{aligned}\mathcal{I}_M [\mathbf{D}\bar{L}(\mathbf{U})] [\mathbf{V}] &= \sum_{m=1}^M \nabla \bar{y}_m(\mathbf{U}) \cdot \mathbf{V} \xi_m \\ &= \sum_{i=1}^H \sum_{m=1}^M \partial_i \bar{y}_m(\mathbf{U}) V_i \xi_m = \sum_{i \in \tau} \sum_{m=1}^M \partial_i \bar{y}_m(\mathbf{U}) V_i \xi_m,\end{aligned}$$

where $\tau \subset \{1, \dots, H\}$ is smallest subset such that above equation is true.

Definition

- Dof mapping $\Phi : \mathbb{R}^H \rightarrow \mathcal{W}_H$, $\mathbf{U} := \{U_i\}_{i=1}^H \mapsto \sum_{i=1}^H U_i \psi_i$.
- Discrete operator $\bar{L} : \mathbb{R}^H \rightarrow \mathcal{W}_H$ given by $\bar{L}[\mathbf{U}] := L \circ \Phi[\mathbf{U}]$.

Derivative of empirical interpolation

Observation

$$\begin{aligned}\mathcal{I}_M [\mathbf{D}\bar{L}(\mathbf{U})] [\mathbf{V}] &= \sum_{m=1}^M \nabla \bar{y}_m(\mathbf{U}) \cdot \mathbf{V} \xi_m \\ &= \sum_{i=1}^H \sum_{m=1}^M \partial_i \bar{y}_m(\mathbf{U}) V_i \xi_m = \sum_{i \in \tau} \sum_{m=1}^M \partial_i \bar{y}_m(\mathbf{U}) V_i \xi_m,\end{aligned}$$

where $\tau \subset \{1, \dots, H\}$ is smallest subset such that above equation is true. **Note:** $\text{card}(\tau) \in \mathcal{O}(M)$ and V_i can be computed efficiently if basis is nodal.

Definition

- Dof mapping $\Phi : \mathbb{R}^H \rightarrow \mathcal{W}_H$, $\mathbf{U} := \{U_i\}_{i=1}^H \mapsto \sum_{i=1}^H U_i \psi_i$.
- Discrete operator $\bar{L} : \mathbb{R}^H \rightarrow \mathcal{W}_H$ given by $\bar{L}[\mathbf{U}] := L \circ \Phi[\mathbf{U}]$.

Non-linear implicit operator

Newton method

Define the defect

$$v_h^{k+1,\nu+1} := u_h^{k+1,\nu+1} - u_h^{k+1,\nu}$$

and solve

$$\mathbf{D}L_I(u_h^{k+1,\nu})[v_h^{k+1,\nu+1}] = -L_I[u_h^{k+1,\nu}] - L_E[u_h^k],$$

for each $k = 0, \dots, K-1$ and each Newton step $\nu = 0, \dots, S(k)$ with
 $u_h^{k+1} := u_h^{k+1,0} := u_h^{k,S(k)}$.

Non-linear implicit operator II

Reduced basis scheme

$$GA(U_N^{k+1,\nu})(\mathbf{a}^{k+1,\nu+1} - \mathbf{a}^{k+1,\nu}) = RHS(\mathbf{a}^{k+1,\nu}, \mathbf{a}^k),$$

with $A(U_N)$ and G defined by

$$(A(U_N))_{m,n} := \sum_{i=1}^M \partial_i y_m(U_N) \varphi_n(x_i) \quad \text{and} \quad (G)_{n,m} = \int_{\Omega} \xi_m \varphi_n.$$

The assembling of $A(U_N)$ costs $\mathcal{O}(MN \cdot MN)$ and multiplication with G has costs of $\mathcal{O}(NMN)$. In addition this is still independent of H .

A posteriori error estimator

Idea

- ▶ Find estimator $\Delta_{N,M}(\mu)$ that bounds error $\|u_h^k(\mu) - u_N^k(\mu)\| \leq \cdot$.
- ▶ It should be computed efficiently during online-phase. \Rightarrow usage in POD-greedy algorithm.
- ▶ Assume there is an $M' > o$ such that empirical interpolation is exact, i.e.

$$L(\mu)[u_N^k] \in \mathcal{W}_{M+M'}$$

A posteriori error estimator (purely explicit scheme)

Residuum

$$R^k := \frac{1}{\Delta t} \left(u_N^{k-1} - u_N^k - \Delta t \mathcal{I}_M \left[L_E(\mu)[u_N^k] \right] \right).$$

Its norm can be efficiently computed by

$$\begin{aligned} \Delta t^2 \|R^k\|_{L^2(\Omega)}^2 = & \\ & \left\| \mathbf{a}^{k-1}(\mu) - \mathbf{a}^k(\mu) \right\|^2 - 2\Delta t \left(\mathbf{a}^{k-1}(\mu) - \mathbf{a}^k(\mu) \right)^t \mathbf{C}_E \mathbf{l}_E(\mu) [\mathbf{a}^k(\mu)] \\ & + \Delta t^2 (\mathbf{l}_E(\mu) [\mathbf{a}^{k-1}(\mu)])^t \mathbf{M} \mathbf{l}_E(\mu) [\mathbf{a}^{k-1}(\mu)] \end{aligned}$$

with reduced matrices C_E and M .

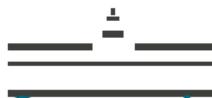
A posteriori error estimator (purely explicit scheme)

Definition

$$\begin{aligned}\Delta_{N,M}^k(\mu) := & \sum_{k'=1}^k \Delta t C_L^{k-k'} \left(\sqrt{\|\tilde{Q}_{M'}^t W^t \tilde{Q}_{M'}\|} \|(\theta_{M+1}^{k'-1}(\mu), \dots, \theta_{M+M'}^{k'-1}(\mu))\| \right. \\ & \left. + \|R^{k'}(\mu)\|_{L^2(\Omega)} \right),\end{aligned}$$

with suitable matrices $Q_{M'}$ and W , such as El-error estimates $\{\theta_{M+m}^{k'}(\mu)\}_{m=1}^{M'}$ given by the linear equation system

$$\sum_{m=1}^{M'} \theta_{M+m}^{k'}(\mu) q_{M+m} = L_E[u_N^{k'}] - \mathcal{I}_M[L_E[u_N^{k'}]]$$



Convection-diffusion equation with parametrized geometry

As a test case we choose a general (maybe nonlinear) convection-diffusion-reaction equation on parametrized geometries.

$$\partial_t u(x, t; \mu) - \nabla \cdot (K \nabla u(x, t; \mu)) - b \cdot \nabla u(x, t; \mu) - r u(x, t; \mu) = 0 \quad \text{in } \Omega(\mu) \times [0, T_{\max}].$$

Convection-diffusion equation with parametrized geometry

The reduced basis space must not depend on the parameter.

Therefore, we introduce a reference geometry $\hat{\Omega}$ and a **diffeomorphic** mapping $\Phi(\mu) : \hat{\Omega} \rightarrow \Omega(\mu)$ for every parameter.

As a test case we choose a general (maybe nonlinear) convection-diffusion-reaction equation on parametrized geometries.

$$\partial_t u(x, t; \mu) - \nabla \cdot (K \nabla u(x, t; \mu)) - b \cdot \nabla u(x, t; \mu) - ru(x, t; \mu) = 0 \quad \text{in } \Omega(\textcolor{red}{X}) \times [0, T_{\max}].$$

Geometry transformation

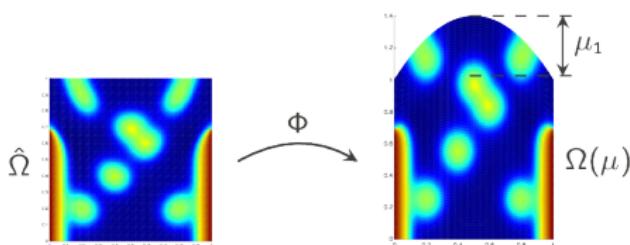
Transformed heat equation

The special case of a transformed linear heat equation with a scalar diffusion factor $k(\mu)$ results in a PDE with (anisotropic) diffusion, convection and a reaction term:

$$\partial_t \hat{u} - k(\mu) \nabla \cdot (GG^t \nabla \hat{u}) + k(\mu) \nabla \cdot (v\hat{u}) - k(\mu)(\nabla \cdot v)\hat{u} = 0 \quad \text{in } \hat{\Omega} \times [0, T_{\max}],$$

with notations

$$\begin{aligned}\hat{x} &:= \Phi^{-1}(x), & \hat{u}(\hat{x}, t) &:= u(\Phi(\hat{x}), t), \\ G(\hat{x}) &:= D\Phi^{-1}|_{\Phi(\hat{x})}, & v(\hat{x}) &:= G(\hat{x})(\nabla_{\hat{x}} \cdot G(\hat{x})).\end{aligned}$$



Finite volume scheme

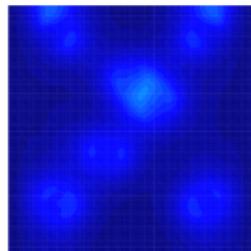
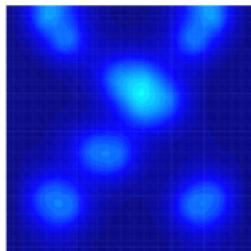
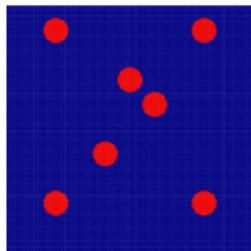
Challenges

- ▶ The geometry transformation introduces a **diffusion tensor**.
- ▶ For non-affine geometry transformations the discrete operators do **not** depend **affinely** on the parameter. ⇒ Empirical interpolation of the purely explicit discretization operator (case 2) or the implicit and explicit operators (case 4).

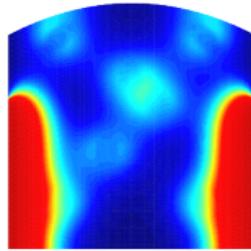
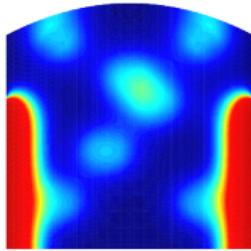
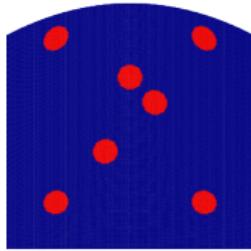
Numerical scheme for transformed heat equation

- ▶ Discretization with a **semi-implicit finite volume scheme** on a structured grid with **gradient reconstruction**. [Drblíková&Mikula, 2007]
- ▶ Implicit discretization of diffusion term.

Results



Solutions for $u = (0, 0)$ at time steps $t = 0.0$, $t = 0.75$, $t = 1.0$



Solutions for $u = (0, 0, 0)$ at time steps $t = 0.0$, $t = 0.75$, $t = 1.0$

Results |

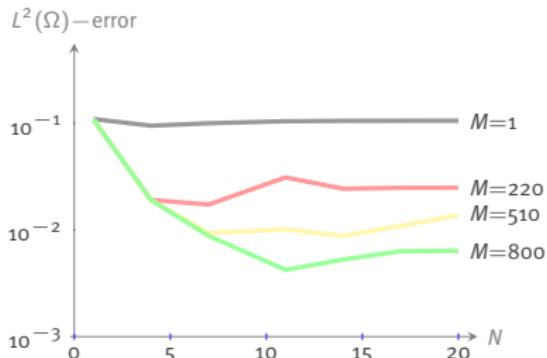


Figure: RB error convergence on 100 test samples

dimension	time [s]
$H = 40000$	24.3675
$N = 7, M = 267$	1.2224
$N = 7, M = 800$	2.0501
$N = 14, M = 267$	1.246
$N = 14, M = 800$	2.104
$N = 20, M = 267$	1.2707
$N = 20, M = 800$	2.1127

Table: average time measurements on 100 test samples

Number of base functions in \mathcal{W}_H : 40000
Time gain factor for online phase: ≈ 10

RB generation with a posteriori error estimates

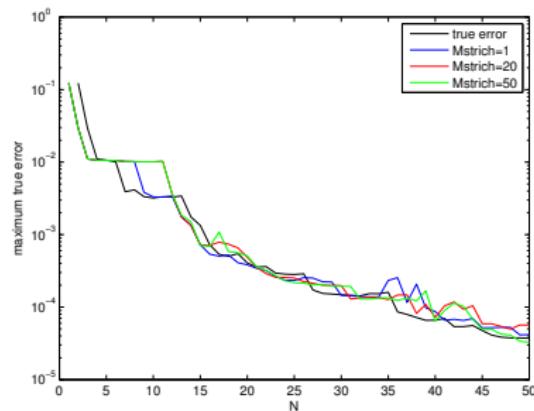


Figure: “true” error decrease for RB generated with different estimators.

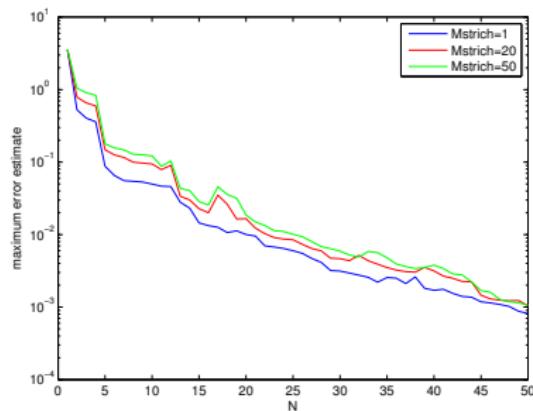
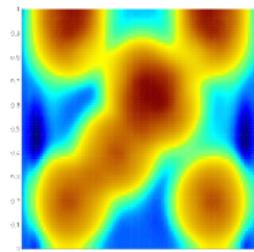
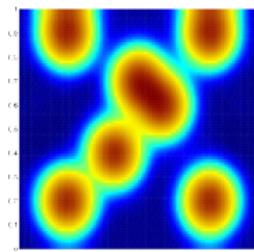
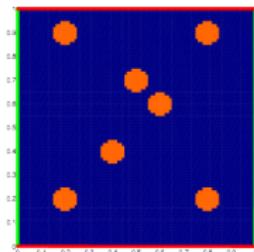
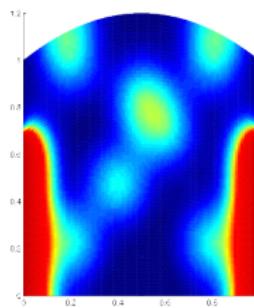
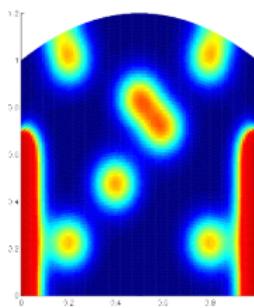
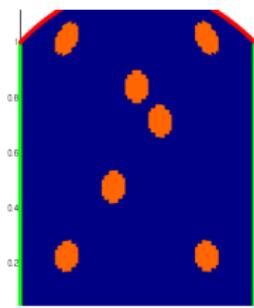


Figure: error estimator decrease for RB generated with different estimators.

Results II (purely explicit FV scheme)



Solutions for $u = (0, 0)$ at time steps $t = 0.0$, $t = 0.45$, $t = 0.9$



Solutions for $u = (0, 2, 0, 2)$ at time steps $t = 0.0$, $t = 0.45$, $t = 0.9$

Results II

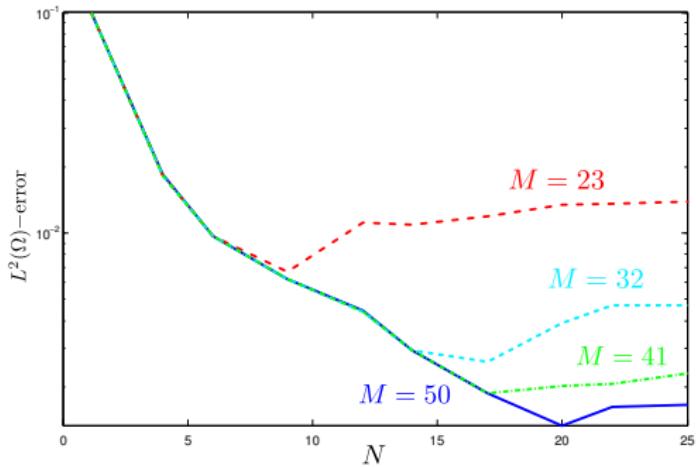


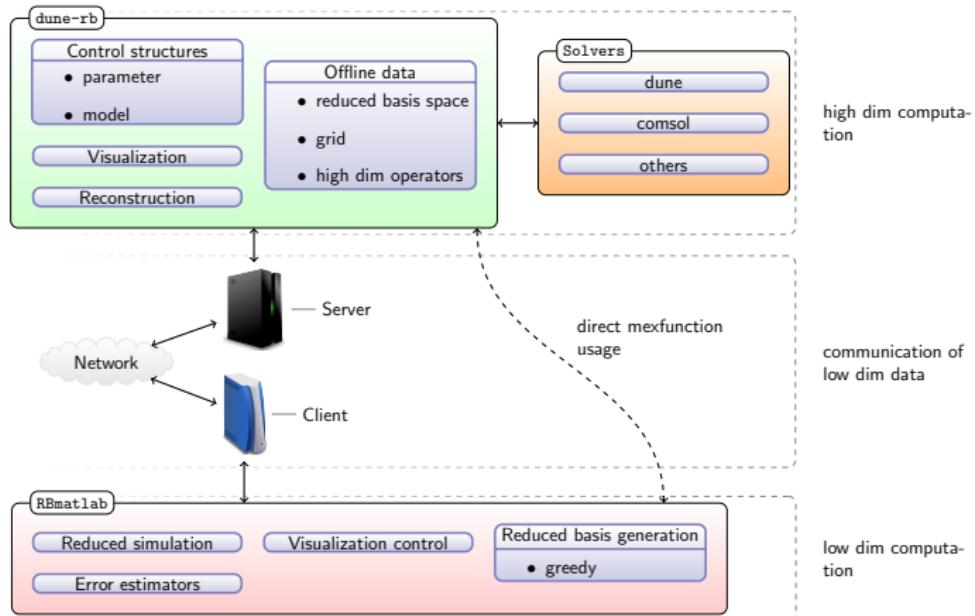
Figure: RB error convergence on 100 test samples for growing basis size N at different CRB dimensions M .

Number of base functions in \mathcal{W}_H : 8000

Number of detailed simulations in offline-phase: 16

Time gain factor for online phase: 5

Software concept





Thanks!

References

-  [Patera, 2006] A. Patera and G. Rozza.
Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations.
Version 1.0, Copyright MIT 2006, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering
-  [Drblíková, 2007] O. Drblíková and K. Mikula,
Convergence Analysis of Finite Volume Scheme for Nonlinear Tensor Anisotropic Diffusion in Image Processing.
SIAM J. Numer. Anal., 46, 27–60, 2007
-  [Haasdonk&Ohlberger, 2008] B. Haasdonk und M. Ohlberger
Reduced Basis Method for Finite Volume Approximations of Parametrized Linear Evolution Equations
M2AN, Math. Model. Numer. Anal. 42, 277–302, 2008
-  [Haasdonk et al., 2008] B. Haasdonk, M. Ohlberger und G. Rozza.
A Reduced Basis Method for Evolution Schemes with Parameter-Dependent Explicit Operators
ETNA, Electronic Transactions on Numerical Analysis, vol. 32, 2008
-  [Barrault et al., 2004] M. Barrault, M., Y. Maday, N. Nguyen and A. Patera,
An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations
C. R. Math. Acad. Sci. Paris Series I, 2004, 339, 667–672
-  [Drohmann et al., 2009] M. Drohmann, B. Haasdonk and M. Ohlberger,
Reduced Basis Method for Finite Volume Approximation of Evolution Equations on Parametrized Geometries
Proceedings of ALGORITMY 2009, 111–120